

The Role of Automation in Software Testing

Bhupinder Paul Singh Sahni

Email: [bhupinder.sahni\[at\]gmail.com](mailto:bhupinder.sahni[at]gmail.com)

Abstract: *The development of software is a complex and expensive process, prone to errors and subsequent failure to meet user requirements. Organizations invest significant resources into ensuring that software products are tested against set criteria, ensuring they are of the best quality before being released to their clients and users. Advances in technology and the constant desire to improve quality have introduced and increased the use of automated testing. Automation Testing has been used for improving the effectiveness of software testing. Testers can focus on critical software features or more complex cases, leaving repetitive tasks to the test automation system. In this Paper, we study what is Automation Testing, its advantages and disadvantages, tests that can be automated and a high - level idea of the decision process Organizations take to Automated Tests.*

Keywords: Software Quality Assurance; Software Testing; Automation Testing; Manual Testing; Decision of Automation Testing

1. Introduction

Software engineering research has two key objectives. First, the reduction of costs and secondly, the improvement of the quality of software products [1]. As software testing represents a significant part of quality costs, the successful introduction of test automation infrastructure has a possibility to combine these two objectives, and to overall improve the software testing processes. Also, with the rapid growth of the acceptance of agile software development in industry, test automation has become a common practice in software development. Testers can focus on critical software features or more complex cases, leaving repetitive tasks to the test automation system. This way it may be possible to use human resources more efficiently, which consequently may contribute to more comprehensive testing or savings in the testing process and overall software development budget [2].

As the speed of software development is increasing today, there is a need for quality and timely testing. But some of the testing tasks are too time - consuming to perform manually. The use of automated test tools to support the test process is proving to be beneficial in terms of product quality and minimizing project schedule and effort. Automated test capabilities continue to increase to keep pace with the growing demand for more rapid and less expensive development for better quality software applications. This paper will study what it means by Automation Testing, its advantages and disadvantages, and testing that can be automated considering the cost of Automated Testing.

2. What is Automation Testing?

Software testing is a systematic process of evaluating software to ensure that it meets specified requirements and functions correctly. The testing work can be divided into manual testing and automated testing. Automation Testing is a software Testing technique that uses Automation Tools to test and control the execution of Test Cases for the software under test. Automated testing requires writing up computer programs called the Automation Test scripts to find bugs or defects. It is an excellent approach to replace the laborious and time - consuming manual testing [3].

Automation is usually applied to running repetitive tasks such as unit testing or regression testing, where test cases are executed every time changes are made [4]. The goal of automation testing is to complete test execution efficiently and in the least amount of time. Test Automation is now applied to more and more complicated software systems in practice [5].

Automated Test Engineers use automated test tools that generate code for developing automation test scripts. The code generated consists of third - generation languages, such as BASIC, C, or C++.

a) Advantages and Disadvantages of Automated Testing

Automation testing offers several advantages and disadvantages, depending on various factors such as the project's requirements, resources, tasks being automated, and the nature of the application being tested. Below are some of the common advantages and disadvantages Organizations experience when Automation Testing is implemented [6].

Some of the advantages of using Automated Testing include:

- **Execution Speed:** Automated Tests can execute test cases much faster than if the tests were to be run manually by a Tester. This enables faster feedback to the Development team if the software is reliable or has bugs.
- **Accuracy:** Automated Tests eliminate the possibility of human errors, thus providing accurate test results.
- **Time saving for Testers:** Once the automation scripts are developed, Testers or Developers just need to schedule them with minimum supervision. It frees up time for the Testers to test and explore new features.
- **Developers can Automate Tests:** Since Automation Tests are usually written in the same programming language as the code has been written, so it is a good idea for Developers to automate the test scenarios and Testers can explore and test areas which are not efficient for Automation.

Some Disadvantages of using Automated Testing are:

- **Cost of Automation:** Even though Automation Testing could be cost - effective for long running project, but it requires initial investment of time and resources to set - up automation framework, writing automation scripts, etc. which can be significant for complex applications.

- False alarms for Failures: Automation Test Scripts might fail because of environment issues, network problems or test script errors, giving false alarms for failures which actually was not correct.
- Continuous maintenance required: Automation Tests require regular maintenance to keep the Automation Test Scripts and automation framework up - to - date with the changes in functionality or the application under testing.
- Fewer Errors detected than Manual Testing: Automation Test Scripts is able to find only those errors which are being written in automation scripts to validate. Many errors in software are detected during exploratory testing, which Automation Scripts are not able to find.

Overall, while automation testing offers numerous benefits in terms of speed, efficiency, and accuracy, it also comes with challenges such as initial setup costs, maintenance overhead, and limitations in test coverage.

b) Tests that can be automated

Even though we have Automation Testing, and we may be able to automate all the tests that are there for the Project, experience has shown that not all tests are worth automating. Some aspects of testing, such as usability testing and exploratory testing, may be better performed manually, where there is less repetition and requires Testers to explore and be more creative in order to find Bugs in the system [7].

Automation Testing can be applied to various types of Software Testing like Unit Testing, Functional Testing, Regression Testing, Integration Testing, Performance Testing and Load Testing. But if the Organization has to start with one testing type that is repeated over and over again and be the best cost - effective solution to the Organization, then they can start with the Automation of Regression Testing or the Unit Testing.

Regression testing involves retesting unchanged parts of the software to ensure that recent changes in the Software haven't adversely affected the existing functionality. Regression testing aims to validate various parts of the software to ensure that all functionalities are working as expected. Regression Testing is best suited for Automation because of its repetitive nature, frequent execution, and comprehensive coverage of Tests for the entire system.

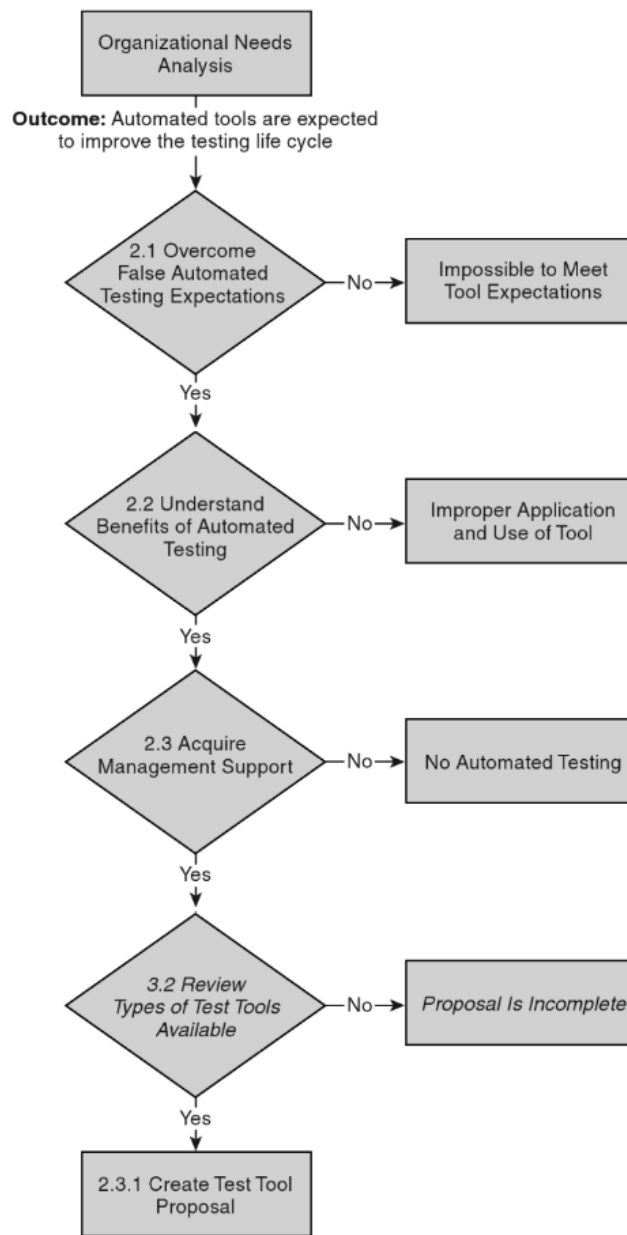
Similarly, Unit Testing can also be considered one of the starting points of automation. Unit testing focuses on testing individual components or units of code in isolation. These Tests can easily be Automated, helps in early detection of bugs and provides fast feedback to the developers, thereby improving the code and product quality.

c) Decision to automate tests

The key factors of using automation are time and cost savings. A survey claims that the cost of software testing is between 30% and 50% of the entire budget, and there is an undeniable requirement for testing methods that can decrease the duration required to guarantee software quality and reliability [8]. To reduce the testing effort without compromising effectiveness and product quality, test automation has been adopted as a popular approach in the software industry. However, because test automation usually requires substantial upfront

investments, automation is not always more cost - effective than manual testing [9].

Efficient Testing ensures a reliable software, but Organization needs to decide of when to use Automation in Testing considering the cost involved. There is always a trade - off between the quality of the software and the cost involved. A structured approach for seeking resource commitment and acquiring management support is described in the figure below [4].

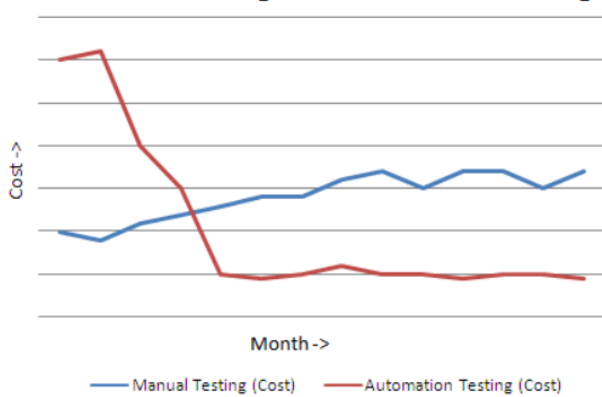


Automated Test Decision Process

But for Organizations to make a decision to adopt Automation Testing for the Projects, depend on various factors like stability of the applications, change in requirements and cost involved. As shown in the graph below, the Cost of Automation Testing is higher at the start of the Project, considering the cost to develop automation framework, getting licenses for automation tools, hiring automation test engineers, etc. But for long term Projects, which are not expected to change much, Automation Testing becomes cost - effective and less costly than the Manual Testing in the long

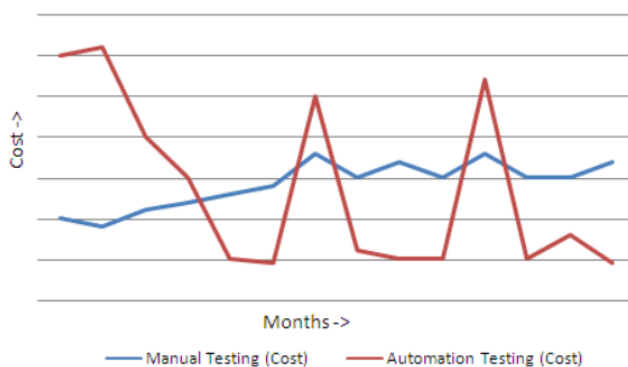
run. Thus, Automation Testing becomes beneficial for the Organizations for long term Projects, where functionalities or applications are not expected to change much [10].

Manual Testing Vs Automation Testing



However, it would be a costlier proposition to automate the functionality or software, which is expected to change significantly soon. As seen in the graph below, the cost of Automation Testing is much higher than Manual Testing in this case where there would be continuous changes in functionalities, resulting in continuous cost to rework on Automation Framework and Automation Test Scripts.

Frequent Change requests



The cost for automation testing is high if you're only running a test once or the requirements are changing frequently, but the cost goes down with each run because the program or the Automation framework and the Automations Test scripts has already been created.

3. Conclusion

Automation testing plays a vital role in modern software development, offering speed, efficiency, and reliability in testing software applications. By automating repetitive and time-consuming test tasks, organizations can streamline their testing processes, improve software quality, and deliver better products to market faster. Test automation has become indispensable to modern software development. But the cost benefit analysis of test automation, what to automate and its impacts on the overall quality and schedule of the software should be studied in detail before approaching any automation technique to be implemented for any Project.

References

- [1] L. J. Osterweil, "Software processes are software too, revisited: an invited talk on the most influential paper of ICSE 9," in Proceedings of the 19th IEEE International Conference on Software Engineering, pp.540–548, Boston, Mass, USA, May 1997.
- [2] R. Ramler and K. Wolfmaier, "Observations and lessons learned from automated testing," in Proceedings of the International Workshop on Automation of Software Testing (AST '06), pp.85–91, Shanghai, China, May 2006.
- [3] D. Kumar, K. Mishra, "The Impacts of Test Automation on Software's Cost, Quality and Time to Market", Procedia Computer Science 79 (2016): 8 - 15.
- [4] E. Dustin, J. Rashka, and J. Paul, Automated Software Testing: Introduction, Management, and Performance, Addison - Wesley, Boston, Mass, USA, 1999.
- [5] Mariani, Leonardo, et al. "The central role of test automation in software quality assurance." Software Quality Journal 25 (2017): 797 - 802.
- [6] Oliinyk, Bohdan, and Vasyli Oleksiuk. "Automation in software testing, can we automate anything we want?." (2019).
- [7] Kasurinen, Jussi, Ossi Taipale, and Kari Smolander. "Software test automation in practice: empirical observations." Advances in Software Engineering 2010 (2010).
- [8] Zhou, Zhi Quan, Arnaldo Sinaga, Willy Susilo, Lei Zhao, and Kai - Yuan Cai. "A cost - effective software testing strategy employing online feedback information." Information Sciences 422 (2018): 318 - 335.
- [9] Garousi, Vahid, and Dietmar Pfahl. "When to automate software testing? A decision-support approach based on process simulation." Journal of Software: Evolution and Process 28.4 (2016): 272 - 285. <https://doi.org/10.1002/smr.1758>
- [10] S. Bhukan. "Automation/Manual Estimates Correlation Part: 2" testingbits. com. Available at: <https://www.testingbits.com/automationmanual-estimates-correlation-part-2/>